Algorithms and Analysis Report
Assignment 2
Path Finding

Joseph Williams: s3569541
Sean Smyth: s3704987

Task A:

The implementation for this task involves getting every possible path from the source to the destination, by putting the source and destination coordinates into a LinkedList, then going through those paths to find the shortest one by distance. Since the terrain cost for each coordinate is the same, the shortest path by route is returned.

Task B:

The implementation for task B was similar to task A, however, now it will compare neighbour's terrain costs. If one neighbour has a higher terrain cost than the other it will go to the neighbour with the lower terrain cost. The shortest path is then computed and returned.

Task C:

Figure 2 shows how the algorithm handles multiple sources and destinations. Like tasks A and B, every possible combination of paths with the lowest cost, from each source to each destination, is put into an adjacency list and the shortest path is then calculated and returned.

Task D:

The implementation for this task works similarly to tasks A, B and C, however instead of searching for the path from source to destination, it searches for the shortest path from a source to a waypoint, then it calculates the path from a waypoint to as many waypoints as needed, treating each pair of waypoints as if they are their own source and destination coordinates, then constantly removing and updating those waypoint coordinates when they are returned. From there it computes the shortest path from the last waypoint to a destination coordinate and then adds them all back to together to make one path (Figure 3).

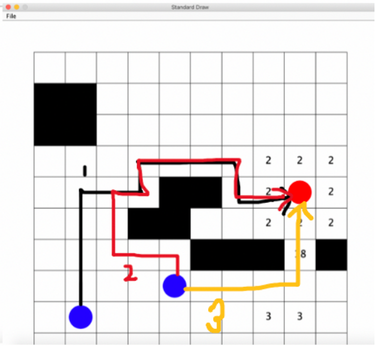

Figure 1: Multiple ways to get to a destination, however the 3rd path will not get chosen unlike in Task A, as its terrain cost exceeds paths 1 and 2. If there was no terrain cost path 3 would be picked.

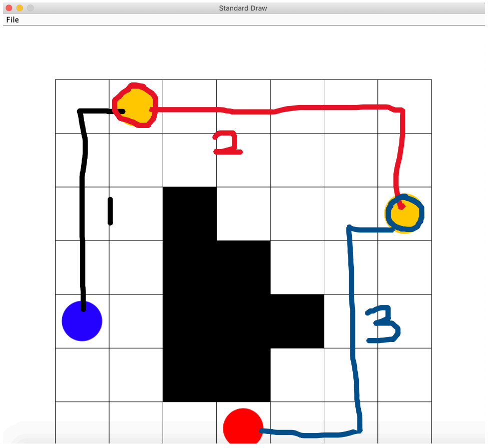

Figure 3: The process to get to a destination via waypoints. Each new waypoint combination is treated as its own source and destination. Therefore, the shortest path from the source to waypoint (path 1) is computed, the waypoint to waypoint is computed (path 2) and the waypoint to destination is computed (path 3). Paths 1, 2 and 3 are then put back together and returned as the final path.

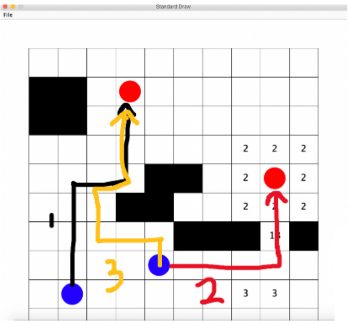

Figure 2: Handling multiple Destinations. Since paths 1 and 3 have the same distance, one will be picked, however path 2 has the highest terrain cost so it will not be picked. If there was no terrain cost path 2 would be picked.